International Journal on Applications in Engineering and Technology
Volume 2: Issue 4: April 2016, pp 13-16. www.aetsjournal.com ISSN (Online) : : 2455-0523

Data Mining Over Encrypted Data For Patient Records

Anand.A, Nithin Chandran.R, Theradath Abhijith, Varun K.K

Abstract-Data Mining has wide applications in many areas such as banking, medicine, scientific research and among government agencies. Classification is one of the commonly used tasks in data mining applications. For the past decade, due to the rise of various privacy issues, many theoretical and practical solutions to the classification problem have been proposed under different security models. However, with the recent popularity of cloud computing, users now have the opportunity to outsource their data, in encrypted form, as well as the data mining tasks to the cloud. Since the data on the cloud is in encrypted form, existing privacy-preserving classification techniques are not applicable. In this paper, the focus is on solving the classification problem over encrypted data

Keywords **-** *m-privacy – K-Nearest Neighbour Classification over Encrypted Data - Data Anonymization - Data Disclosure Property – Encryption using Message Digest5 (MD5) – Random Generator*.

I. INTRODUCTION

Data mining techniques and applications are very much needed in the cloud computing paradigm. As cloud computing is penetrating more and more in all ranges of business and scientific computing, it becomes a great area to be focused by data mining. "Cloud computing denotes the new trend in Internet services that rely on clouds of servers to handle tasks. Data mining in cloud computing is the process of extracting structured information from unstructured or semi-structured web data sources. The data mining in Cloud Computing allows organizations to centralize the management of software and data storage, with assurance of efficient, reliable and secure services for their users."

II.

As Cloud computing refers to software and hardware delivered as services over the Internet, in Cloud computing data mining software is also provided in this way. The main effects of data mining tools being delivered by the Cloud are:

The customer only pays for the data mining tools that he needs – that reduces his costs since he doesn't have to pay for complex data mining suites that he is not using exhaustive; The customer doesn't have to maintain a hardware infrastructure, as he can apply data mining through a browser – this means that he has to pay only the costs that are generated by using Cloud computing. Using data mining through Cloud computing reduces the barriers that keep small companies from benefiting of the data mining instruments.

"Cloud Computing denotes the new trend in Internet services that rely on clouds of servers to handle tasks. Data mining in cloud computing is the process of extracting structured information from unstructured or semi-structured web data sources. The data mining in Cloud Computing allows organizations to centralize the management of software and data storage, with assurance of efficient, reliable and secure services for their users." The implementation of data mining techniques through Cloud computing will allow the users to retrieve meaningful information from virtually integrated data warehouse that reduces the costs of infrastructure and storage.

The relationship between data mining and cloud is worth to discuss. Cloud providers use data mining to provide clients a better service. If clients are unaware of the information being collected, ethical issues like privacy and individuality are violated. This can be a serious data privacy issue if the cloud providers misuse the information. Again attackers outside cloud providers having unauthorized access to the cloud, also have the opportunity to mine cloud data. In both cases, attackers can use cheap and raw computing power provided by cloud computing to mine data and thus acquire useful information from data. As cloud is a massive source of centralized data, data mining gives attackers a great advantage in extracting valuable information and thus violating clients' data privacy.

II K-NN ALGORITHM

In pattern recognition, the k-Nearest Neighbors algorithm (or k-NN for short) is a non- parametric method used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space. The output depends on whether k-NN is used for classification or regression:

(i).In k-NN classification, the output is a class membership. An object is classified by amajority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If k = 1, then the

object is simply assigned to the class of that single nearest neighbor.

(ii).In k-NN regression, the output is the property value for the object. This value is the average of the values of its k nearest neighbors.

K-NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification. The k-NN algorithm is among the simplest of all machine learning algorithms.

Both for classification and regression, it can be useful to assign weight to the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. For example, a common weighting scheme consists in giving each neighbor a weight of 1/d, where d is the distance to the neighbor.

The neighbors are taken from a set of objects for which the class (for k-NN classification) or the object property value (for k-NN regression) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required.

A shortcoming of the k-NN algorithm is that it is sensitive to the local structure of the data. The algorithm has nothing to do with and is not to be confused with k-means, another popular machine learning technique.

The training examples are vectors in a multidimensional feature space, each with a class label. The training phase of the algorithm consists only of storing the feature vectors and class labels of the training samples.

In the classification phase, k is a user-defined constant, and an unlabeled vector (a query or test point) is classified by assigning the label which is most frequent among the k training samples nearest to that query point.

A commonly used distance metric for continuous variables is Euclidean distance. For discrete variables, such as for text

classification, another metric can be used, such as the overlap metric (or Hamming distance). In the context of gene expression microarray data, for example, k-NN has also been employed with correlation coefficients such as Pearson and Spearman. Often, the classification accuracy of k-NN can be improved significantly if the distance metric is learned with specialized algorithms such as Large Margin Nearest Neighbor or Neighborhood components analysis.

A drawback of the basic "majority voting" classification occurs when the class distribution is skewed. That is, examples of a more frequent class tend to dominate the prediction of the new example, because they tend to be common among the k-nearest neighbors due to their large number. One way to overcome this problem is to weigh the classification, taking into account the distance from the test point to each of its k nearest neighbors. The class (or value, in regression problems) of each of the k nearest points is multiplied by a weight proportional to the inverse of the distance from that point to the test point. Another way to overcome skew is by abstraction in data representation. For example in a self-organizing map (SOM), each node is a representative (a center) of a cluster of similar points, regardless of their density in the original training data. K-NN can then be applied to the SOM.

IIIM-PRIVACY

The technique of m-privacy, which guarantees that the anonym zed data satisfies a given privacy constraint against any group of up to m colluding data providers. Finally, we present a data provider-aware anonymization algorithm with adaptive m-privacy checking strategies to ensure high utility and m-privacy of anonymized data with efficiency.

Our goal is to publish an anonymized

view of the integrated data such that a data recipient including the data providers will not be able to compromise the privacy of the individual records provided by other parties. M-Privacy helps in protecting anonymized data against m-adversary with respect to privacy constraint. The proposed modelprovides a competent approach to achieve enhanced privacy for collaborative data publishing. M-privacy techniques assure that the anonymized data fulfils a given privacy constraint against any range of m-colluding data providers (where m can be varied between certain ranges 1 to m).

In existing many techniques are introduced. For example, k-anonymity prevents identity disclosure attacks by requiring each equivalence group, records with the same quasi identifier values, to contain at least k records. Representative constraints that prevent attribute disclosure attacks include l- diversity, which requires each equivalence group to contain at least l "well-represented" sensitive values, and t-closeness, which requires the distribution of a sensitive attribute in any equivalence group to be close to its distribution in the whole population. They can attempt to infer additional information about data coming from other providers by analyzing the data received during the anonymization.

A trusted third party (TTP) or Secure Multi- Party Computation (SMC) protocols can be used to guarantee there is no disclosure of intermediate information

during the anonymization. However, either TTP or SMC do not protect against data providers to infer additional information about other records using the anonymized data and their own data. Hence in the proposed system we introduced m-privacy technique to overcome these challenges.

MESSAGE DIGEST5 (MD5) ALGORITHM

The MD5 function is a cryptographic algorithm that takes an input of arbitrary length and produces a message digest that is 128 bits long. The digest is sometimes also called the "hash" or "fingerprint" of the input. MD5 is used in many situations where a potentially long message needs to be processed and/or compared quickly. The most common application is the creation and verification of digital signatures.

EXECUTION:

II.

The MD5 algorithm first divides the input in blocks of 512 bits each. 64 Bits are inserted at the end of the last block. These 64 bits are used to record the length of the original input. If the last block is less than 512 bits, some extra bits are 'padded' to the end. Next, each block is divided into 16 words of 32 bits each. These are denoted as M_0 ... M_{15}.

MD5 helper functions

THE BUFFER

MD5 uses a buffer that is made up of four words that are each 32 bits long. These words are called A, B, C and D. They are initialized as

Word A: 01 23 45 67

Word B: 89 ab cd ef Word C: fe dc ba 98 Word D: 76 54 32 10

II.

THE TABLE

MD5 further uses a table K that has 64

elements. Element number i is indicated as K_i. The table is computed beforehand to speed up

the computations. The elements are computed using the mathematical sin function:

$$K_i = abs (sin (i + 1)) * 2^{32}$$

I. FOUR AUXILIARY FUNCTIONS

In addition MD5 uses four auxiliary functions that each take as input three 32-bit words and produce as output one 32-bit word. They apply the logical operators and,or, not and xor to the input bits.

F(X, Y, Z) = (X and Y) or (not(X) and Z)

G(X, Y, Z) = (X and Z) or (Y and not (Z)) H(X, Y, Z) = X xor Y xor Z

I(X, Y, Z) = Y xor (X or not (Z))

II. PROCESSING THE BLOCKS

The contents of the four buffers (A, B, C and

D) are now mixed with the words of the input, using the four auxiliary functions (F, G, H and I). There are four *rounds*, each involves 16 basic *operations*.

EXISTING SYSTEM

Existing work on privacy-preserving data mining (PPDM) (either perturbation or secure multi-party computation (SMC) based approach) cannot solve the DMED problem. Perturbed data do not possess semantic security, so data perturbation techniques cannot be used to encrypt highly sensitive data. Also the perturbed data do not produce very accurate data mining results. Secure multi-party computation based approach assumes data are distributed and not encrypted at each participating party.

K-NEAREST NEIGHBOUR CLASSIFICATION OVER SEMANTICALLY SECURE ENCRYPTED DATA

Privacy-preserving k-nn classification protocol, denoted by ppknn

PROPOSED SYSTEM

To effectively solve the DMED problem assuming that the encrypted data are outsourced to a cloud. Specifically, we focus on the classification problem since it is one of the most common data mining tasks. Because each classification technique has their own advantage, to be concrete, this paper concentrates on executing the k-nearest neighbor classification method over encrypted data in the cloud computing environment. The proposed system can be implementing in any of the application.

In the proposed system, patient health records are preserved by hiding the sensitive data. Data disclosure property is achieved here based on the type of users. Data anonymization technique is used. Data anonymization is type of information sanitization i.e. process of removing the sensitive data information.

III. EXPERIMENTAL ANALYSIS

Representative constraints that prevent attribute disclosure attacks include l-diversity, which requires each equivalence group to contain at least l "well-represented" sensitive values, and t-closeness, which requires the distribution of a sensitive attribute in any equivalence group to be close to its distribution in the whole population. They can attempt to infer additional information about data coming from other providers by analyzing the data received during the anonymization. A trusted third party (TTP) or Secure Multi-Party Computation (SMC) protocols can be used to guarantee there is no disclosure of intermediate information during the anonymization. However, either TTP or SMC do not protect against data providers to infer additional information about other records using the anonymized data and their own data.

I. VIII CONCLUSION & FUTURE WORK

Considering various factors such as personal security and data access data mining over encrypted data using m-privacy is very prominent. The unauthorized data access can be prevented and hence people can make sure that their personal information's are safe from unauthorized usage

As the Innovations along with the Technologies increases, it overcomes certain limitations that had been recorded from the proposed System, It will be a Very helpful and Serving Factor, if the Proposed Application is being implemented in Public Environment. If the External User Cannot Access the Encrypted Data or any Similar Consequences Occur, Biometrics Can be Implemented So as to Gain Access to The Encrypted Data and their Limitations Can be Overcame.

REFERENCES

[1] P. Mell and T. Grance, "The NIST definition of cloud computing (draft)," NIST Special Publication, vol. 800, p. 145, 2011.

[2] S. De Capitani di Vimercati, S. Foresti, and P. Samarati, "Managing and accessing data in the cloud: Privacy risks and approaches," in Proc.

7th Int. Conf. Risk Security Internet Syst., 2012, pp. 1–9.

[3] P. Williams, R. Sion, and B. Carbunar, "Building castles out of mud: Practical access pattern privacy and correctness on untrusted storage," in Proc. 15th ACM Conf. Comput. Commun. Security, 2008, pp. 139–148.

[4] P. Paillier, "Public key cryptosystems based on composite degree residuosity classes," in

Proc. 17th Int. Conf. Theory Appl. Cryptographic Techn., 1999, pp. 223–238.

[5] B. K. Samanthula, Y. Elmehdwi, and W. Jiang, "k-nearest neighbor classification over semantically secure encrypted relational data," eprint arXiv: 1403.5001, 2014.

[6] C. Gentry, "Fully homomorphic encryption using ideal lattices," in Proc. 41st Annu. ACM

Sympos. Theory Compute. 2009, pp. 169–

178.

[7] C. Gentry and S. Halevi, "Implementing gentry's fully-homomorphic encryption scheme," in Proc. 30th Annu. Int. Conf. Theory Appl. Cryptographic Technical.: Adv. Cryptol., 2011, pp. 129–148.

[8] A. Shamir, "How to share a secret," Community. ACM, vol. 22, pp. 612–613,

1979.

[9] D. Bogdanov, S. Laur, and J. Willemson, "Sharemind: A framework for fast privacy- preserving computations," in Proc. 13th Eur. Symp. Res. Comput. Security: Comput. Security, 2008, pp. 192–206.

[10] R. Agrawal and R. Srikant, "Privacy- preserving data mining,"ACM Sigmod Rec., v